Accredited

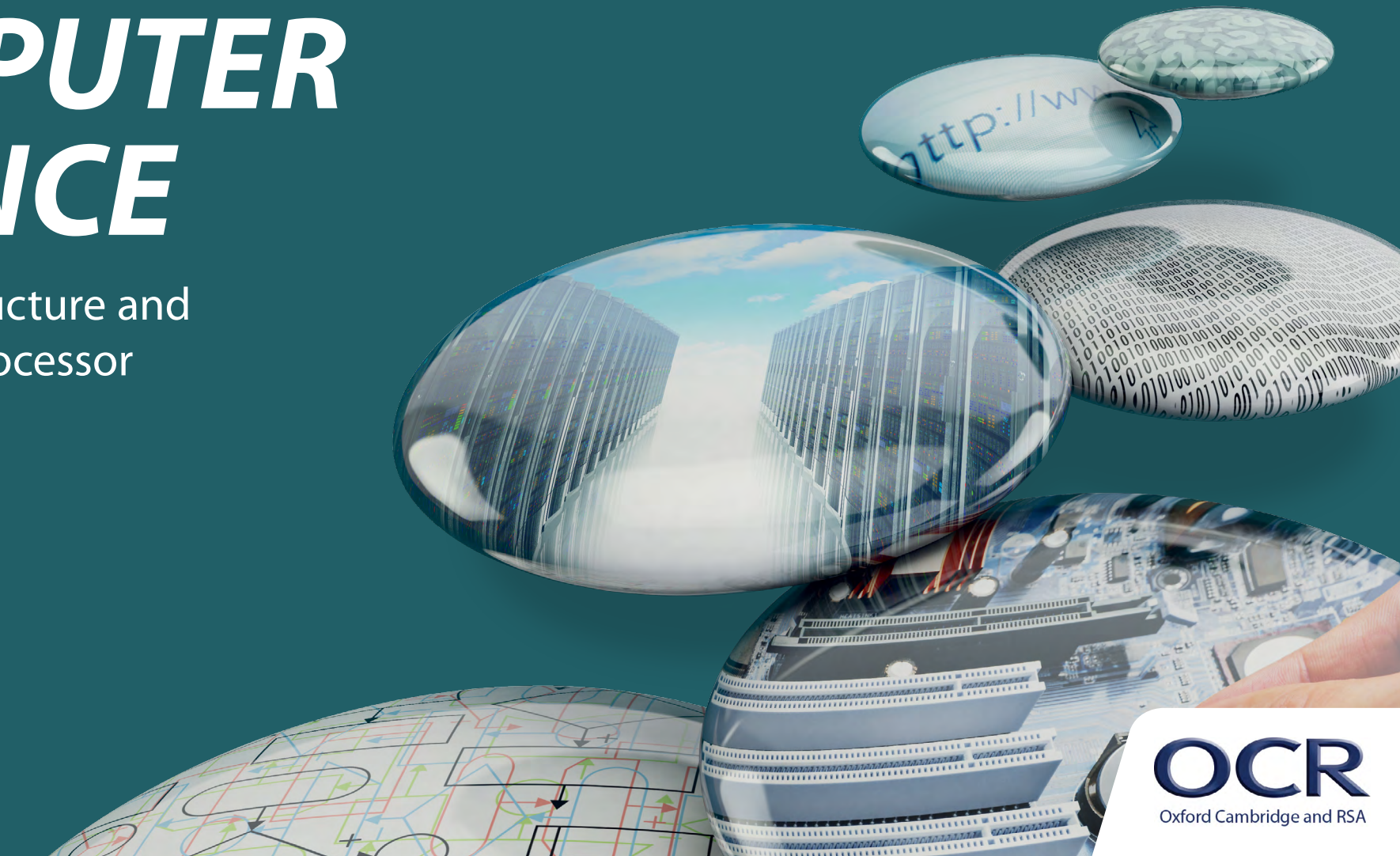
A LEVEL
Delivery Guide

H446

COMPUTER SCIENCE

Theme: 1.1.1 Structure and Function of a Processor

June 2015



OCR
Oxford Cambridge and RSA

We will inform centres about any changes to the specification. We will also publish changes on our website. The latest version of our specification will always be the one on our website (www.ocr.org.uk) and this may differ from printed versions.

Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee.
Registered in England. Registered company number 3484466.

Registered office: 1 Hills Road
Cambridge
CB1 2EU

OCR is an exempt charity.

CONTENTS

Introduction

Delivery guides are designed to represent a body of knowledge about teaching a particular topic and contain:

- Content: a clear outline of the content covered by the delivery guide;
- Thinking Conceptually: expert guidance on the key concepts involved, common difficulties students may have, approaches to teaching that can help students understand these concepts and how this topic links conceptually to other areas of the subject;
- Thinking Contextually: a range of suggested teaching activities using a variety of themes so that different activities can be selected that best suit particular classes, learning styles or teaching approaches.

If you have any feedback on this Delivery Guide or suggestions for other resources you would like OCR to develop, please email resources.feedback@ocr.org.uk.

KEY

Click to view associated resources within this document.

Click here — Click to view external resources

AS Level only — AS Level content only

Curriculum Content

a) The Arithmetic and Logic Unit; ALU, Control Unit and Registers (ProgramCounter; PC, Accumulator; ACC, MemoryAddress Register; MAR, Memory Data Register; MDR, Current Instruction Register; CIR). Busses: data, address and control: How this relates to assembly language programs

b) The Fetch-Decode-Execute Cycle, including its effect on registers

c) The factors affecting the performance of the CPU: clock speed, number of cores

d) Cache

e) The use of pipelining in a processor to improve efficiency

f) Von Neumann, Harvard and contemporary processor architecture.

Thinking Conceptually

Approaches to teaching the content	Resources
This topic starts by exploring in detail the CPU/Processor at a working level. Students will learn about the various parts of the CPU and how they interact and function to process instructions and 'compute'.	Click here
There are a selection of interactive sites which explain and provide animations for the various component parts of a processor: http://www.eastaughs.fsnet.co.uk/cpu/index.htm	
This video demonstrates how the CPU works: https://www.youtube.com/watch?v=cNN_tTXABUA	Click here
The Fetch-Decode-Execute Cycle is a model to demonstrate what the parts of the Processor are doing and the effect that this has on the internal memory registers. There are many videos showing this cycle in action which will support students' understanding of the process: https://www.youtube.com/watch?v=04UGopESS6A https://www.youtube.com/watch?v=J9I2erViGqs	Click here
Students need to understand the Fetch-Decode-Execute Cycle at the registry and component level. Little Man Computer is free software which models a simple von Neumann architecture computer; it has all of the basic features of a modern computer. Little Man Computer can be programmed in machine or assembly code, which will enable you to run your programs. This does require software to be installed first, although students may be familiar with the interface from GCSE: http://www.yorku.ca/sychen/research/LMC/	Click here
The topic covers the factors that affect CPU performance; students are required to identify what they are as well as explain the impact that each will have on performance. This can be achieved by comparing several PROCESSOR specifications and professional reviews of the performance. Benchmark testing is a useful tool to compare CPU performance across a range of brands and technologies. Students can then discuss and analyse, in relation to the factors, why one processor outperforms another. This resource: https://www.cpubenchmark.net/ allows students to compare and contrast a range of CPU benchmarks or look for a specific model.	Click here
Depending on what access rights students have to computers, they could predict the performance of a computer, based on the factors, and install the free software to benchmark test it: http://novabench.com/	Click here
A further technique that improves the efficiency of the processor is pipelining. This is a commonly used structure that improves the performance of a processor by using the output from one execution as the input for another. This approach is common in manufacturing and assembly lines. This site gives a good overview: http://en.wikipedia.org/wiki/Pipeline_(computing)	Click here

Thinking Conceptually

Approaches to teaching the content	Resources
Pipelining is a technique used to speed up the fetch-decode-execute cycle; this is important when comparing the differences between and benefits of CISC and RISC in 1.1.2. Types of Processor. Finally, this topic traces back to the original concepts of processor architecture developed by Von Neumann and Harvard. These contemporary processor architectures are also historical and many resources provide a detailed guide to how the architectures were developed and how they differ from each other: http://en.wikipedia.org/wiki/Harvard_architecture http://www.princeton.edu/~achaney/tmve/wiki100k/docs/Harvard_architecture.html http://www.eeherald.com/section/design-guide/dg.html Students will gain an insight into how these architectures have developed into today's modern-day processors, supporting discussions around future advancements.	Click here Click here Click here
Common misconceptions or difficulties students may have	**Resources**
Students who have studied GCSE Computing will have an understanding of the CPU and its function. Students may consider a CPU/Processor as three elements, the ALU, CU and Cache. Within these are more elements which function together to 'process' a set of instructions.	
Conceptual links to other areas of the specification – useful ways to approach this topic to set students up for topics later in the course	**Resources**
This unit content will support elements 1.1.2 Types of Processor, where students explore types of processors, and 1.2.1 Systems Software, where students study Interrupts, the role of Interrupts and Interrupt Service Routines (ISR) and their role within the fetch-decode-execute cycle.	

Thinking Contextually

Activities	Resources
Activity 1 • Cut out the cards in Learner Resource 1 and place them in a box or bag. • Students take it in turns to pull one out. Depending on resources, the students could use plasticine to model the word or a concept that represents the word. • Students could draw the word instead or use actions to explain the word without talking. • To add an element of competition, students could be split into groups and play against each other, for example two students have the same word and have to use the plasticine to create something that represents the word which enables their team to guess correctly. • After each correct guess, the students should be asked how they guessed it; this can be an opportunity to ask further questions or clarify misunderstanding.	Learner Resource 1
Activity 2 *Part 1* • Using the cards in Learner Resource 1, cut them up, give one to each student or give them all to a group of students. The task is to put them into the correct order of the FETCH actions and the EXECUTE actions. Students should justify why they have ordered the actions and explain what is happening. *Part 2* • Make an animation of the Fetch-Decode-Execute Cycle. Learner Resource 2 shows a basic explanation. • Use these sites to support your knowledge and check that your animation is correct: - http://www.teach-ict.com/as_as_computing/ocr/H447/F453/3_3_3/fetch_execute_cycle/miniweb/index.htm - http://en.wikibooks.org/wiki/A-level_Computing/AQA/Computer_Components,_The_Stored_Program_Concept_and_the_Internet/Machine_Level_Architecture/The_Fetch%E2%80%93Execute_cycle_and_the_role_of_registers_within_it	Learner Resource 2 Click here Click here

Learner Resource 1 Activity 1 cards

See page 8

The Arithmetic and Logic Unit	Data and Address Control	ALU, Control Unit and Registers	Program Counter; PC	Accumulator	Memory Address Register
Memory Data Register	Von Neumann architecture	Harvard architecture	The Fetch-Decode-Execute Cycle	Pipelining	CPU
Performance	Assembly language	Number of Cores	Current Instruction Register	Busses	Clock Speed

Learner Resource 2 Fetch-Decode-Execute Cycle

See page 8

Fetch

- The Program Counter (PC) contains the address of the next instruction to be fetched.
- The address contained in the PC is copied to the Memory Address Register (MAR).
- The instruction is copied from the memory location contained in the MAR and placed in the Memory Buffer Register (MBR).
- The entire instruction is copied from the MBR and placed in the Current Instruction Register (CIR).
- The PC is incremented so that it points to the next instruction to be fetched.

Execute

- The address part of the instruction is placed in the MAR.
- The instruction is decoded and executed.
- The processor checks for interrupts (signals from devices or other sources seeking the attention of the processor) and either branches to the relevant interrupt service routine or starts the cycle again.


We'd like to know your view on the resources we produce. By clicking on the 'Like' or 'Dislike' button you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click 'Send'. Thank you.

If you do not currently offer this OCR qualification but would like to do so, please complete the Expression of Interest Form which can be found here:
http://www.ocr.org.uk/qualifications/expression-of-interest/

OCR customer contact centre
General qualifications
Telephone 01223 553998
Facsimile 01223 552627
Email general.qualifications@ocr.org.uk